

31759-190545
Akihiro OKUMURA et al

日 本 国 特 許 庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2002年11月28日

出 願 番 号

Application Number:

特願2002-345885

[ST.10/C]:

[JP 2002-345885]

出 願 人

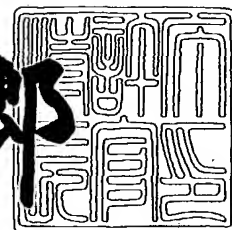
Applicant(s):

沖電気工業株式会社

2003年 4月 1日

特 許 庁 長 官
Commissioner,
Japan Patent Office

太田 信一郎



出証番号 出証特2003-3022445

【書類名】 特許願
【整理番号】 KN002562
【提出日】 平成14年11月28日
【あて先】 特許庁長官 太田 信一郎 殿
【国際特許分類】 G06F 15/00
【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会
社内

【氏名】 奥村 晃弘

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会
社内

【氏名】 池野 篤司

【特許出願人】

【識別番号】 000000295

【氏名又は名称】 沖電気工業株式会社

【代表者】 篠塚 勝正

【代理人】

【識別番号】 100090620

【弁理士】

【氏名又は名称】 工藤 宣幸

【先の出願に基づく優先権主張】

【出願番号】 特願2002-187622

【出願日】 平成14年 6月27日

【手数料の表示】

【予納台帳番号】 013664

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

特2002-345885

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9006358

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 情報処理装置および方法

【特許請求の範囲】

【請求項1】 複数の領域を含む所定の構造化文書の中から、主要な領域を判定する情報処理装置において、

前記構造化文書中の各領域または、各領域の管理情報を時系列に複数回、取得する読み込み部と、

当該読み込み部が取得した領域または管理情報を記憶する記憶部と、

当該読み込み部が取得した対応する領域の間または管理情報の間で比較を行い、当該比較結果に基づいて各領域の更新の有無を検査する比較検査部と、

当該比較検査部による検査結果の履歴をもとに、各領域ごとに、所定の更新頻度対応情報を算出する更新頻度算出部と、

当該更新頻度対応情報に基づいて、前記構造化文書中の複数の領域の中から主要な領域を判定する主要領域判定部とを備えたことを特徴とする情報処理装置。

【請求項2】 請求項1の情報処理装置において、

前記構造化文書中の各領域を、画面表示上の領域間の境界を示す境界情報に基づいて分割する境界分割部を備え、

前記読み込み部は、当該境界分割部と連携することによって、前記各領域または管理情報の読み込みを行うことを特徴とする情報処理装置。

【請求項3】 請求項1の情報処理装置において、

前記構造化文書中の各領域が、予め領域間の論理的な構造を示す領域間構造を定義されている場合には、その領域間構造を利用して各領域を識別する領域管理部を備え、

前記読み込み部は、当該領域管理部と連携することによって、前記各領域または管理情報の読み込みを行うことを特徴とする情報処理装置。

【請求項4】 請求項1の情報処理装置において、

前記比較検査部は、

前記読み込み部が今回、取得した領域と当該領域に対応する前回、取得した領

域とを比較することによって更新の有無を検査することを特徴とする情報処理装置。

【請求項5】 請求項1の情報処理装置において、

前記比較検査部は、

前記読み込み部が今回、取得した管理情報と当該管理情報に対応する前回、取得した管理情報とを比較することによって更新の有無を検査することを特徴とする情報処理装置。

【請求項6】 請求項1の情報処理装置において、

前記領域または管理情報を、誤り検出のため、より小さなサイズの変換データに変換する変換部を備え、

当該変換部が出力する変換データを、前記記憶部に記憶することを特徴とする情報処理装置。

【請求項7】 請求項1の情報処理装置において、

前記更新頻度算出部は、

前回の更新頻度対応情報と、今回、前記比較検査部が出力した検査結果に基づいて、今回の更新頻度対応情報を算出することを特徴とする情報処理装置。

【請求項8】 複数の領域を含む所定の構造化文書の中から、主要な領域を判定する情報処理方法において、

読み込み部が、前記構造化文書中の各領域または、各領域の管理情報を時系列に複数回、取得し、

当該読み込み部が取得した領域または管理情報を、記憶部が記憶し、

比較検査部が、当該読み込み部の取得した対応する領域の間または管理情報の間で比較を行い、当該比較結果に基づいて各領域の更新の有無を検査し、

当該比較検査部による検査結果の履歴をもとに、更新頻度算出部が、各領域ごとに所定の更新頻度対応情報を算出し、

当該更新頻度対応情報に基づいて、主要領域判定部が、前記構造化文書中の複数の領域の中から主要な領域を判定することを特徴とする情報処理方法。

【発明の詳細な説明】

【0001】

本発明は情報処理装置および方法に関し、例えば、WWW (World Wide Web) サイトから構造化文書を取得する場合などに適用し得るものである。

【0002】

【従来の技術】

WWWサイトに存在する構造化文書を取得し閲覧するためのツールとして、WWWブラウザがある。一般的に、構造化文書はその文書のページのレイアウト、文字の大きさなどを柔軟に指定することができるようになっている。特に、図1のように、タイトル(領域A)、他の構造化文書へのリンク(領域B)、本文(領域C)、など、ページがいくつかの領域(フレーム)に分割されて、WWWブラウザに表示されるような構造化文書が多く見られる。WWWブラウザを用いて、このような構造化文書から必要な情報を得るためには、ユーザは、目的の構造化文書のURLを指定し、その文書がWWWブラウザ上に表示された後に、文書をスクロールしながら目視により検索したり(人手による検索)、あるいは文字列検索機能を利用するといった作業を行なう必要がある。例えば、図1の領域Cが、ユーザの必要とする文書であったとし、こういった構造化文書が多数ある場合には、そのユーザが必要とする情報のみを複数の構造化文書から自動的にスクラップし、1つの文書にまとめてユーザに提示することが、人手による作業を簡略化する上で望ましくなる。このようなWWW情報抽出システムが、次の特許文献1に示されている。

【0003】

【特許文献1】

特開平10-187753号公報

【0004】

【発明が解決しようとする課題】

しかしながら、上記におけるWWW情報抽出システムでは、ユーザが構造化文書中で自分が必要とするデータの開始箇所と終了箇所をあらかじめ手入力により指定することが必要である。このため、大量の構造化文書に対して実施するにはユーザの操作負担が大きく現実的ではなかった。

【0005】

【課題を解決するための手段】

かかる課題を解決するために、第 1 の発明では、複数の領域を含む所定の構造化文書の中から、主要な領域を判定する情報処理装置において、（１）前記構造化文書中の各領域または、各領域の管理情報を時系列に複数回、取得する読み込み部と、（２）当該読み込み部が取得した領域または管理情報を記憶する記憶部と、（３）当該読み込み部が取得した対応する領域の間または管理情報の間で比較を行い、当該比較結果に基づいて各領域の更新の有無を検査する比較検査部と、（４）当該比較検査部による検査結果の履歴をもとに、各領域ごとに、所定の更新頻度対応情報を算出する更新頻度算出部と、（５）当該更新頻度対応情報に基づいて、前記構造化文書中の複数の領域の中から主要な領域を判定する主要領域判定部とを備えたことを特徴とする。

【0006】

また、第 2 の発明では、複数の領域を含む所定の構造化文書の中から、主要な領域を判定する情報処理方法において、（１）読み込み部が、前記構造化文書中の各領域または、各領域の管理情報を時系列に複数回、取得し、（２）当該読み込み部が取得した領域または管理情報を、記憶部が記憶し、（３）比較検査部が、当該読み込み部の取得した対応する領域の間または管理情報の間で比較を行い、当該比較結果に基づいて各領域の更新の有無を検査し、（４）当該比較検査部による検査結果の履歴をもとに、更新頻度算出部が、各領域ごとに所定の更新頻度対応情報を算出し、（５）当該更新頻度対応情報に基づいて、主要領域判定部が、前記構造化文書中の複数の領域の中から主要な領域を判定することを特徴とする。

【0007】

【発明の実施の形態】

（Ａ）実施形態

以下、本発明にかかる情報処理装置および方法の実施形態について説明する。

【0008】

（Ａ－１）第 1 の実施形態の構成

本実施形態で、主要領域を判定し抽出する機能を有する領域処理部 25（図 1

6 参照) の機能は、通信機能を有するパソコンその他の情報処理装置によって実現され得、WWWサーバ側に配置すること等も可能であるが、ここでは、通信端末(クライアント)側に配置する場合を例に説明する。

【0009】

本実施形態にかかる通信システム10の全体構成例を図15に示す。

【0010】

図15において、当該通信システム10は、ネットワーク11と、通信端末12と、WWWサーバ13とを備えている。

【0011】

このうちネットワーク11は、LAN(ローカルエリアネットワーク)などであってもかまわないが、ここでは、インターネットであるものとする。

【0012】

WWWサーバ13は、通信端末12からの要求(HTTPリクエスト)を受信すると、その要求に応じた応答(HTTPレスポンス)としてWWWページを構成するファイルなどを返送する機能を持つサーバである。多くの場合、WWWサーバ13は、予め生成したWWWページ等を蓄積しておくためのデータベース(図示せず)、そのデータベースを直接管理するデータベースサーバを伴う。また、これらWWWサーバ13やデータベースサーバなどの周辺には、ルータやファイアウォールなどの各種のネットワーク機器やDNSサーバなどのサーバ類が配置されてWWWサイトを構成するのが普通である。

【0013】

通信端末12は上述した領域処理部25を備えた情報処理装置で、具体的には、ネットワーク機能を有するパーソナルコンピュータなどであってよい。本実施形態の構成上、当該通信端末12は、WWWページを閲覧するためのプログラムであるWWWブラウザB1(図16参照)を搭載していることが必要である。

【0014】

当該通信端末12の内部構成例を図16に示す。

【0015】

(A-1-1) 通信端末の内部構成例

図 1 6 において、当該通信端末 1 2 は、通信部 2 0 と、制御部 2 1 と、操作部 2 2 と、記憶部 2 3 と、表示部 2 4 と、領域処理部 2 5 とを備えている。

【 0 0 1 6 】

このうち通信部 2 0 は、前記ネットワーク 1 1 を介して WWW サーバ 1 3 と通信する機能を持つ部分である。

【 0 0 1 7 】

制御部 2 1 は、ハードウェア的には当該通信端末 1 2 の中央処理装置 (CPU) に相当する部分であり、ソフトウェア的には、オペレーティングシステム (OS) や上述した WWW ブラウザ B 1 などに相当する部分である。

【 0 0 1 8 】

操作部 2 2 は当該通信端末 1 2 のユーザ U 1 が操作して制御部 2 1 に指示を伝えるための部分で、例えば、キーボードや、ポインティングデバイスなどを有する。

【 0 0 1 9 】

表示部 2 4 は例えば液晶ディスプレイなどの表示画面を有する部分である。ユーザ U 1 が WWW ページを閲覧する場合には、WWW ブラウザ B 1 が WWW ページのタグを解釈し処理した結果として、WWW ページの内容が当該表示部 2 4 に画面表示され、ユーザ U 1 に閲覧され得る。このとき画面表示される WWW ページは、一例として、図 1 に示した DP 1 であってよい。DP 1 のようなフレームのページ (フレームページ) を表示するためには、WWW ブラウザ B 1 はフレームに対応したものであることを要する。フレームとは、図 1 の各領域 A ~ C に画面表示されている内容のことではなく、その内容を収容している枠のことを指す。図 1 のフレームページは、例えば、図 4 (A) ~ (D) の HTML ファイルをもとに生成される。

【 0 0 2 0 】

通常、1 つの WWW ページは、基本となる 1 つの HTML ファイルと、必要に応じて 1 または複数の各種ファイル (画像ファイルなど) から構成されるが、DP 1 のようなフレームページは、これよりファイル数も多く、複雑な構造を有する。

【 0 0 2 1 】

すなわちフレームページは、そのWWWページ全体の構成（すなわち、フレーム構造）を規定するHTMLファイル（フレーム規定ファイル）と、内容として各フレームに配置される複数のHTMLファイルが最低限、必要であり、これに加えて、各HTMLファイルにリンクされた各種ファイル（画像ファイルなど）が適宜、追加されることになる。

【 0 0 2 2 】

したがって、簡単のために各種ファイルがないと仮定しても、図1に示したフレームページDP1は、図4（A）のHTMLファイル（フレーム規定ファイル）と、領域Aに配置される図4（B）のHTMLファイルと、領域Bに配置される図4（C）のHTMLファイルと、領域Cに配置される図4（D）のHTMLファイルの、合計4つのHTMLファイルから構成されることになる。

【 0 0 2 3 】

通常のWWWページは、1つのHTMLファイルの内部でのみ構造化されているが、フレームページでは、各HTMLファイルの内部で構造化されているだけでなく、1つのフレームページに含まれる複数のHTMLファイルのあいだでも構造化されている。

【 0 0 2 4 】

なお、図1では、各領域A～Cのあいだの境界線（スクロールバーなども含む）L1、L2が表示されているが、実際のフレームページでは、視覚的な効果などとの関連で、意図的にこのような境界線が表示されないようにし、異なる領域間でまったく地の色が同じになるようにしたり、まったく切れ目のない連続的な背景模様を表示したりすることも多い。したがって、境界線の表示の有無は、領域分割（フレーム構造）の本質とは関係ない。

【 0 0 2 5 】

フレーム構造、すなわち、1つの画面をいくつのフレームに分割し、各フレームの辺の長さの割合（この割合は、各フレームの面積に対応）をどのように設定するか等（境界線の表示、非表示も含む）は、前記フレーム規定ファイル（例えば、DP11）の記述によって決まる。

【0026】

当該フレームページDP1の閲覧を希望する場合、ユーザU1が通信端末12のWWWブラウザB1に入力するのは、フレーム規定ファイルDP11のURL（ここでは、URL11とする）である。したがってこのとき、当該フレーム規定ファイルDP11の返送を要求するHTTPリクエストが通信端末12からWWWサーバ13へ送信され、そのHTTPレスポンスとして、各種HTTPヘッダ（エンティティヘッダも含む）とともに、エンティティボディとして当該フレーム規定ファイルDP11が返送される。

【0027】

エンティティボディ、すなわち、HTMLファイルや画像ファイルなどのファイルの返送を要求する場合、HTTPリクエストはGETメソッドを使用するGETリクエストになる。

【0028】

その他のHTMLファイルDP12～DP14は、当該フレーム規定ファイルDP11が通信端末12に受信されたあと、当該フレーム規定ファイルDP11中の記述（URL12～URL14）に基づいて、WWWブラウザB1が順次、自動的に送信する各HTTPリクエストに応じた各HTTPレスポンスとして、通信端末12に受信される。

【0029】

そして、これら4つのHTMLファイル进行处理し、整形した結果として、例えば、図1に示すような画面が、前記表示部24に表示されることになる。

【0030】

ここで、HTMLファイルDP12のURLを前記URL12とし、HTMLファイルDP13のURLを前記URL13とし、HTMLファイルDP14のURLを前記URL14とすると、URL12は図4（A）の行TG12中の「title.html」であり、URL13は行TG13中の「menu.html」であり、URL14は行TG14中の「main.html」である。

【0031】

通常、1つのフレームページを構成する複数のHTMLファイルは、フレーム

規定ファイルと同じWWWサーバ（ここでは、13）に置かれ、しかも同じフォルダ内に置かれるため、FQDNを含まないこのようなローカルなURL（ここでは、ファイル名だけから成るURL）によって指定することができる。

【0032】

必要ならば、1つのフレームページを構成する複数のHTMLファイルを別なWWWサーバに置くこともでき、その場合には、例えば、前記URL12～URL14の全部または一部が、FQDNを含むURLに変わる。なお、ユーザU1によって入力され得るフレーム規定ファイルDP11を指定する前記URL11は、FQDNを含むURLであることは当然である。

【0033】

このURL11は、ユーザU1が操作部22を用いて行う手作業によって入力され得るほか、例えば、前記特許文献1に記載されたオートパイロットツールのようなソフトウェアを利用し、予めユーザU1が設定した日時や時間間隔で自動的に入力させることができる。

【0034】

図1のようなメニューを有するフレームページの場合、メニューのファイルが図4（C）に示すような内容（ターゲット（target）がすべて、main（領域Cのフレーム名）になっている）であると、メニュー内でユーザU1がハイパーリンクを選択することにより、そのハイパーリンクのリンク先のファイルが、領域Cに置き換わって表示されることになる。

【0035】

したがって、典型的な使用方法では、メニューを有するフレームページの場合、他のフレームの内容（ここでは、領域A、領域B）は同じでターゲットのフレームの内容（領域C）だけが置き換わった複数のWWWページが重層的に存在しているとみることができる。

【0036】

図16に示す前記記憶部23は、例えば、RAM（ランダムアクセスメモリ）などの揮発性記憶装置や、ハードディスクなどの不揮発性記憶装置によって構成される。

【 0 0 3 7 】

通信端末 1 2 が前記 WWW サーバ 1 3 から WWW ページを構成する各ファイル（例えば、前記 DP 1 1 や DP 1 2 など）を受信すると、それらのファイルは、当該記憶部 2 3 のハードディスク上に確保されたキャッシュ領域に一時的に蓄積される。

【 0 0 3 8 】

当該キャッシュ領域は、通常、WWW ブラウザ B 1 の管理下に置かれ、WWW ブラウザ B 1 から自由にアクセスすることが可能である。

【 0 0 3 9 】

当該キャッシュ領域におけるファイルの記憶は可能なかぎり維持されるが、キャッシュ領域の記憶容量には上限があるため、この上限を越えて新たな WWW ページを閲覧し、新たなファイルを蓄積するときには、すでに記憶されているファイルを例えば古いものから順番に削除することによって、必要な記憶容量を確保することになる。

【 0 0 4 0 】

例えば URL の入力などにより、ユーザ U 1 が操作部 2 2 から WWW ページの閲覧を指示した場合、その URL で指定される WWW ページに関連するファイルがキャッシュ領域に記憶されていれば WWW ブラウザ B 1 は、インターネット 1 1 経由ではなく、当該キャッシュ領域から当該ファイルを取得し、表示部 2 4 に当該ファイルの内容を画面表示する。これによりネットワーク 1 1 上の通信トラフィックや、WWW サーバ 1 3 側の負荷の増大を抑制し、ユーザ U 1 から見た応答時間（指示を出してから WWW ページが表示されるまでの時間）を短縮することができる。

【 0 0 4 1 】

前記制御部 2 1 に接続された領域処理部 2 5 は、主要領域を判定し抽出する機能を有する部分であるが、その内部構成は、例えば、図 2 に示すようになる。

【 0 0 4 2 】

主要領域とは、WWW ページ上の複数の領域のうちユーザ（ここでは、U 1）にとって最も重要であると推定できる 1 つの領域のことである。本実施形態では

、基本的に、更新の頻度が最も高い領域が主要領域であるものと想定している。したがって、例えば、図1に示したフレームページの場合なら、領域A～Cのうち、ユーザU1にとって最も重要であると推定できる（最も更新の頻度が高い）いずれか1つの領域が主要領域となる。また、メニューを有するフレームページの典型的な使用方法に対応して、ターゲットのフレームの内容（例えば、領域C）だけが置き換わった複数のフレームページごとに、主要領域を決定するようにしてもよい。

【0043】

（A-1-2）領域処理部の内部構成例

図2において、当該領域処理部25は、読み込み部101と、バッファ部102と、分割部103と、分割結果記憶部104と、領域内容比較部105と、更新頻度算出部106と、更新頻度記憶部107と、判定部108とを有する。

【0044】

読み込み部101は、定期的にあるいは不定期（ユーザU1が読み込みを指示した時など）に指定されたURL（例えば、前記URL11）のフレームページに関する情報N10をバッファ部102に読み込む部分である。この読み込みは、記憶部23内の前記キャッシュ領域などから行うようにしてもよいが、通常は、インターネット11経由でWWWサーバ13から行うことになる。

【0045】

このようなインターネット11経由の読み込みを可能にするため、当該読み込み部101は、前記WWWブラウザB1の一部として機能するか、前記WWWブラウザB1と連携するか、あるいは、WWWブラウザB1と独立したHTTPクライアントとしての機能を備える必要がある。

【0046】

読み込み部101が読み込んだ時点のファイルの内容は、一例として、図5に示す複合ファイルSP1のようなものであってもよい。

【0047】

図5において、複合ファイルSP1は、図4（A）～（D）に示すHTMLファイルDP11～DP14の内容を合成したものとなっている。ただしフレーム

規定ファイルDP11の内容は、4つの部分PT1～PT4に分割されて配置されている。

【0048】

WWWサーバ13側でファイルDP11～DP14が1つのフレームページに対応するファイルであることを管理しておき、フレーム規定ファイルDP11に対するHTTPリクエストが受信された時点で4つのファイルDP11～DP14をもとに1つの合成ファイルSP1を生成して返送するようにすれば、合成ファイルSP1のようなファイルを1つのHTTPレスポンスのエンティティボディとして返送することも可能であるが、WWWサーバ13の負荷や、キャッシュシステムとの親和性などを考慮すると、不利であるため、WWWサーバ13側でそのような処理は行わないほうが望ましい。

【0049】

WWWサーバ13側でそのような処理を行わない場合、図1に示すフレームページDP1を閲覧するには、全部で4回のHTTPリクエストが通信端末12から送信され、それにこたえて、順次、4回のHTTPレスポンスがWWWサーバ13から返されることになり、そのうち3回のHTTPレスポンスのエンティティボディが、ファイルDP12～DP14である。

【0050】

したがって、SP1のようなファイルは、多くの場合、通信端末12がすべてのファイルDP11～DP14を受信したあとの処理（整形処理）によって、はじめて得られるものである。

【0051】

バッファ部102は、読み込み部101が読み込んだ内容や分割部103が処理した結果を一時的に記憶する部分である。当該バッファ部102や、分割結果記憶部104、更新頻度記憶部107などの記憶機能を実現するための記憶資源は、前記記憶部23とは別個に確保してもよいが、前記記憶部23内で確保するようにしてもよい。

【0052】

分割部103は、バッファ部102に読み込まれたフレームページを解析する

ことによって、予め指定された文書構造に基づいて、バッファ部102の記憶内容を領域に分割する部分である。ただしフレームページの場合、フレームページ上の各領域（例えば、領域A～領域C）は異なるHTMLファイルに対応し、最初から（すなわち、前記整形処理を行う前は）別ファイルに分割されているため、通常のファイル管理を行うことのできるファイルシステムを搭載していれば、分割は省略できる可能性がある。ファイル管理は通常、OSの役割であるから、分割部103はOSに対して、例えば、バッファ部102上の次のファイルを渡すように要求しその応答としてのファイルを受け取る処理を繰り返すだけでよい。

【0053】

ただし前記WWWブラウザB1と当該領域処理部25（読み込み部101）のインタフェースなどとの関係で、整形処理前のファイルDP11～DP14を受け取ることができず、整形処理後の合成ファイルSP1しか受け取ることができない場合などには、分割が必要になる。

【0054】

分割は、例えば図5に示す合成ファイルSP1から、部分PT1～PT4を除外して図6に示す3つのファイルDP12、DP13、DP14を得る処理である。

【0055】

分割部103が分割した（OSから受け取った）ファイル（例えば、DP12～DP14）は、分割結果記憶部104へ供給する。

【0056】

分割結果記憶部104は、分割部103から前回、供給を受けたファイルN11を記憶しておく部分である。前回、供給を受けたファイルN11は、今回、供給を受けバッファ部102内に格納した該当するファイルN12と比較するために記憶するものであるから、前回と今回で、各ファイルの対応関係がわかるような形で記憶しておく必要があることは当然である。例えば、前回の領域CのHTMLファイルと、今回の領域CのHTMLファイルDP14が、他のHTMLファイル（例えば、領域BのHTMLファイルDP13など）と混同されないよう

にしておく必要がある。

【0057】

この対応付けには、例えば、各HTMLファイルのファイル名（例えば、URL12～14）を利用することもでき、システム内部で各HTMLファイルに付与した固有のファイル識別子（領域番号）を利用することもできる。

【0058】

領域内容比較部105は、分割結果記憶部104が記憶している前回、供給を受けた各ファイルN11とバッファ部102が記憶している今回、供給を受けた各ファイルとを、ファイルごとに（すなわち、領域ごとに）内容を比較して更新の有無を検出する部分で、検出結果として、更新有無情報N16を出力する。更新とは、ファイルの内容の一部または全部が、追加、削除、または、変更されることである。

【0059】

更新頻度算出部106は、当該更新有無情報N16の内容と前回までの更新頻度から所定の式に基づいて今回の更新頻度Sを算出し更新頻度Sに応じた更新頻度情報N17を出力する部分である。更新頻度Sの算出は、ファイルごと（領域ごと）に行う。更新頻度は、ファイルの更新の頻度が増加するほどその値が増加するものであってもよいが、本実施形態では、ファイルの更新の頻度が増加するほどその値が減少するものを用いる。

【0060】

また、前記式としては様々な形の式を用いることが可能であるが、本実施形態では、指数平均による次の式（1）を用いるものとする。

【0061】

$$S = S_0 \alpha + P (1 - \alpha) \quad \dots (1)$$

ここで、 S_0 は、前回までの更新頻度であり、 α は $0 < \alpha < 1$ の定数であるが、ここでは $\alpha = 0.8$ とする。また、Pは得点であり、100または0のいずれかの値を持つ。後述するように、更新が行われた場合には得点Pを0とし、更新が行われていない場合には100とするため、更新頻度Sの値は、（ S_0 を正の値とすると）更新が行われるたびに小さくなる。

【0062】

なお、GETリクエストに対するHTTPレスポンスでは、エンティティボディ（ファイル）のほかに、各種HTTPヘッダも返送されるため、更新頻度算出部106の処理において、HTTPヘッダ中の情報を利用することも望ましい。

【0063】

例えば、HTTPヘッダの1つであるエンティティヘッダに含まれる更新日情報や有効期限情報を利用することも望ましい。

【0064】

更新日情報は、WWWサーバ13側でファイルが更新された日付を示す情報である。

【0065】

また、有効期限情報は、ファイルを（前記キャッシュ領域やキャッシュサーバなどに）キャッシュする際の有効期限を設定するための情報で、提供するファイルの内容の最新性を維持するために使用される。有効期限情報は、通常、ファイルの作成者（WWWページの作成者）など、WWWサーバ13側の都合に合わせて設定される。有効期限が経過すれば、たとえ記憶容量に余裕があっても、通信端末12（あるいは、通信端末12とWWWサーバ13のあいだに介在する前記キャッシュサーバ）は、当該ファイルをキャッシュ領域から削除し、オリジナルのファイルをWWWサーバ（ここでは、13）から取得することになる。したがって、有効期限を短く設定すれば高頻度の更新が行われても最新の内容を持つファイルをユーザ（ここでは、U1）に提供できる可能性が高まる利点があるが、有効期限を短く設定するとキャッシュの利点が小さくなりWWWサーバ13側の負荷も増大するため、通常は、できるだけ長い有効期限を設定しようとする。

【0066】

このため、有効期限情報は、WWWページの作成者などが、当該ファイルをどの程度の頻度で更新する予定であるか、あるいは、どの程度の頻度で更新される（例えば、電子掲示板（CGI掲示板）など、第3者によってファイルの更新が行われるWWWページの場合（ただし、普通、電子掲示板はキャッシュ禁止とする））と予想しているかを示す情報（更新予定情報）であるとみることにもできる

。WWWページの作成者等は、通常、ユーザU1などからは知ることのできない、自身の予定やWWWページの性質などを熟知しているはずだから、更新予定情報としての有効期限情報の信頼度は高いといえる。

【0067】

前記更新頻度算出部106がこのような有効期限情報を利用する方法には様々なものがあり得るが、例えば、更新頻度Sを算出する際、有効期限情報が示す有効期限が短いほど、更新頻度Sが高頻度側に変更されるように重み付けを施すことも望ましい。

【0068】

当該更新頻度算出部106から当該更新頻度Sに応じた更新頻度情報N17を受け取る更新頻度記憶部107は、領域ごとに、当該更新頻度情報N17を記憶する部分である。

【0069】

判定部108は、更新頻度記憶部107の記憶する更新頻度情報N17に基づいて最も頻繁に内容が変化している領域を主要領域と判定し、バッファ部102から該当領域の内容を取り出してそれを出力する部分である。主要領域の出力先には様々なものが考えられるが、例えば、WWWブラウザB1であってよい。WWWブラウザB1が当該主要領域を表示部24に画面表示するために必要であれば、前記フレーム規定ファイルDP11も、当該主要領域とともにWWWブラウザB1へ出力するとよい。

【0070】

一方、前記HTTPリクエストにこたえてHTTPレスポンスを返すWWWサーバ13の内部構成例は、図17に示す。

【0071】

(A-1-3) WWWサーバの内部構成例

図17において、当該WWWサーバ13は、通信部30と、制御部31と、記憶部32とを備えている。

【0072】

このうち通信部30は前記通信部20に対応し、制御部31は前記制御部21

に対応し、記憶部 32 は前記記憶部 23 に対応するので、その詳しい説明は省略する。

【0073】

ただし制御部 31 は、WWWブラウザ（B1 など）を搭載することはなく WW
Wサーバソフトを搭載している。

【0074】

また、WWWサーバ 13 が、予め生成した WWW ページ等を蓄積しておくため
のデータベースを伴う場合には、必要に応じて、DBMS も、当該制御部 31 に
搭載され得る。

【0075】

さらに、前記フレームページ DP1 を提供するため、当該記憶部 32 には、少
なくとも、HTML ファイル DP11 ~ DP14 が格納されている。

【0076】

以下、上記のような構成を有する本実施形態の動作について、図 3、図 7、図
8 の各フローチャートを参照しながら説明する。

【0077】

図 3 は更新頻度情報 N17 などの情報を更新するためのフローチャートで、S
101 ~ S104 の各ステップを備えている。同様に、図 7 は更新頻度情報 N1
7 などをもとに、主要領域の内容を判定し出力するためのフローチャートで、S
101、S102、S105 の各ステップを備えている。

【0078】

また、図 8 は更新頻度 S の算出またはその近傍の処理を示すフローチャートで
、S151 ~ S160 の各ステップを備えている。図 8 のフローチャートは、図
3 中のステップ S103 の詳細な動作を示すものである。

【0079】

（A-2）第 1 の実施形態の動作

ユーザ U1 が操作部 22 を操作することによって、または、前記オートパイロ
ットツールなどの機能によって、URL11 が WWW ブラウザ B1 に入力される
と、WWW ブラウザ B1 が当該 URL11 に対応した HTTP リクエスト（GE

トリクエスト)を送信する。

【0080】

ただし前記読み込み部101に前記WWWブラウザB1と独立したHTTPクライアントとしての機能を持たせる場合には、この入力WWWブラウザB1ではなく、当該読み込み部101に対して行うことは当然である。

【0081】

通信端末12から送信されたこのHTTPリクエストをインターネット11経由で受信すると、WWWサーバ13(およびサーバOS)は、前記記憶部32から当該URL11によって指定されるフレーム規定ファイルDP11を取り出し、当該フレーム規定ファイルDP11をエンティティボディとして含むHTTPレスポンスを返送する。

【0082】

フレームページDP1を構成するその他のHTMLファイルDP12~DP14は、上述したように、当該フレーム規定ファイルDP11が通信端末12に受信されたあと、当該フレーム規定ファイルDP11中の記述(URL12~URL14)に基づいて、WWWブラウザB1が順次、自動的に送信する各HTTPリクエストに応じた各HTTPレスポンスのエンティティボディとして、通信端末12に受信されるから、これらを、前記読み込み部101が読み込むことになる。

【0083】

ただしWWWサーバ13側で前記合成ファイルSP1を生成する場合には、この合成ファイルSP1をエンティティボディとして含むHTTPレスポンスが返されることになる。

【0084】

図3に示すように、通信端末12の領域処理部25に含まれる読み込み部101は単独で、またはWWWブラウザB1と連携して当該ファイル(DP11またはSP1)を読み込んでバッファ部102へ格納することになる(S101)。単独で読み込む場合には、HTTPレスポンスのエンティティボディがそのまま読み込み部101に読み込まれ、WWWブラウザB1と連携して読み込む場合に

は、WWWブラウザB1側で処理を施された結果が、読み込み部101に読み込まれることになる。

【0085】

いずれの場合も、このとき読み込み部101に読み込まれるファイルには、図4(A)に示すフレーム規定ファイルDP11と図5に示す合成ファイルSP1の2通りがあり得る。

【0086】

このとき読み込み部101に読み込まれるファイルが合成ファイルSP1である場合には分割する(S102)が、フレーム規定ファイルDP11である場合には、分割する必要はない。

【0087】

ただしフレーム規定ファイルDP11である場合には、上述した3つのHTMLファイルDP12～DP14を取得するために、逐次、通信端末12からHTTPリクエストを送信する必要があり、順次、受信されるHTTPファイルを読み込む必要がある。

【0088】

合成ファイルSP1を用いる場合も用いない場合も、各HTMLファイルDP12～DP14は、前記ファイル名のほか、前記領域番号によって識別し対応付けることが可能である。

【0089】

ここでは一例として、HTMLファイルDP12には領域番号1を付与して領域1とし、HTMLファイルDP13には領域番号2を付与して領域2とし、HTMLファイルDP14には領域番号3を付与して領域3とする。

【0090】

次に、ステップS103では、前記領域内容比較部105が、今回、読み込んだ(必要に応じて、分割された)ファイルと前回、読み込んだファイルとを比較して更新の有無を検出し、それをもとに更新頻度算出部106が更新頻度Sを算出し、さらに、算出した更新頻度Sを更新頻度記憶部107が記憶する。

【0091】

そして、今回、読み込んだ（分割された）ファイルDP12～DP14の内容は、前回、読み込んだファイルに置き換える形で、分割結果記憶部104に格納し、次の新たな読み込みに備える（S104）。

【0092】

このようなステップS101～104の処理は、必要に応じて、複数回、繰り返される。わずかの回数（例えば、1回）、更新が検出されただけでは、偶然、そのタイミングでそのファイルが更新されただけかもしれないので、このような偶然性を排除して真に更新の頻度が高いか否かを見極め所期の効果を得るためには、繰り返しの回数が多いほうが望ましく、更新の有無の検出は長期間にわたって行うほうが望ましいことは当然である。

【0093】

なお、必要に応じて、前記有効期限情報（更新予定情報）を活用することにより、繰り返しの回数が比較的すくなく、更新の有無を検出する期間が比較的に短い場合でも、偶然性を排除できる可能性がある。

【0094】

また、繰り返しの回数が少なく、更新の有無の検出期間が短い場合、最近、更新が行われたファイルほど主要領域として選ばれる確率が高まる。ユーザU1にとって最も重要な領域は、最も近い過去に更新された領域であると想定することができるとするなら、繰り返しの回数が少なく、更新の有無の検出期間が短くても、重要な領域を選択できる。

【0095】

前記ステップS103の詳細を示す図8では、前記領域番号を利用して、各ファイルDP12～DP14を識別している。

【0096】

図8において、ステップS152～S159は、バッファ部102に格納されている各ファイルDP12～DP14に付与された領域番号をもとに、ファイルごとに（領域番号ごとに）繰り返される。

【0097】

このうちステップS152では、前回、読み込んだ（前回、分割した）ファイ

ルと、今回、読み込んだ（今回、分割した）ファイル、すなわち、分割結果記憶部104に格納されているファイルとバッファ部102に格納されているファイルを、前記領域番号をもとに対応付けて比較する（S153）。

【0098】

比較の結果、内容が同じであれば、すなわちファイルが更新されていなければ、ステップS153はYes側に分岐し、前記式（1）の得点Pに100を設定し（S155）、当該式（1）に基づいて更新頻度Sを算出する（S157）。ここで算出された更新頻度Sにより、前回、算出され更新頻度記憶部107に記憶されていた更新頻度 S_0 が置換される。

【0099】

比較の結果、前記領域1の内容が前回と同じであり、領域1の前の更新頻度 S_0 を $S_0 = 73$ とすると、図9に示すように、前記式（1）にしたがって、領域1の今回の更新頻度Sは、 $78 (\equiv 73 \times 0.8 + 100 \times (1 - 0.8))$ となる。

【0100】

また、前記領域2の内容が前回と同じであり、領域2の前の更新頻度 S_0 を $S_0 = 73$ とすると、領域1の今回の更新頻度Sも、78となる。

【0101】

一方、前記ステップS153における比較の結果、内容が同じでなければ、すなわちファイルが更新されていなければ、ステップS153はNo側に分岐し、分割数が同じであるか否かを検査する（S154）。分割数とは、前記フレームページDP1を構成するフレーム規定ファイルDP11以外のファイルの数であるから、このステップS154では、前記フレーム構造（特に、フレームの数）そのものが変化しているか否かを検査していることになる。

【0102】

なお、フレーム構造の変化の有無は、このように分割数を調べなくても、直接、前記フレーム規定ファイルDP11の記述内容を解析（例えば、「<FRAME src」という文字列の数をカウントする）することによっても、検査することが可能である。

【0103】

また、フレーム構造の概念には上述したように、フレームの数だけでなく、各フレームの辺の長さの割合なども含まれるが、ステップS154では、フレームの数だけを問題にしている。したがって、フレーム規定ファイルDP11の記述が変化し各フレームの辺の長さの割合などが変化したとしても、フレームの数と同じであればステップS154はYes側に分岐する。もちろん、必要ならば、フレーム規定ファイルDP11の記述が少しでも変化したらステップS154をNo側に分岐させるようにしてもよい。

【0104】

例えば、行TG12～TG14に記述されているURL12～URL14のいずれかが変化した場合には、同じファイルの内容が更新されたのではなく、そのフレームに配置されるファイルが別ファイルに変更されたことを意味するため、ステップS154をYes側に分岐させるようにしてもよい。

【0105】

一例として、それまで領域1～領域3の3つの領域しか存在しなかったフレームページDP1に、図10に示す領域4（領域番号4を付与した領域）のような新たな領域が出現した場合には、ステップS154はNo側に分岐し、当該領域4の更新頻度Sには0が設定され（S158）、処理は前記ステップS159に進む。この場合、当該0は更新頻度の初期値であり、フレームページDP1のフレーム構造そのものが変化してしまったため、比較すべき領域がなくなったとみなして、更新頻度を初期値にリセットしたものである。

【0106】

また、分割数が同じであってステップS154がYes側に分岐すると、その領域には、前記得点Pに0を設定して（S156）、処理をステップS157に進める。

【0107】

ここで、図9に示すように、領域3に対する処理でステップS154がYes側に分岐し、領域3の前の更新頻度 S_0 を $S_0 = 46$ とすると、前記式(1)にしたがって、領域3の今回の更新頻度Sは、 $37 (\equiv 46 \times 0.8 + 0 \times (1$

-0. 8)) となる。

【0108】

以上のようにして、更新頻度記憶部107に記憶されている各領域1～領域3の更新頻度Sの更新は、1または複数回おこなわれるが、更新頻度Sをもとに、前記主要領域を判定し、当該主要領域を出力する場合は、図7のようなフローチャートとなる。

【0109】

図7において、図3と同じ符号S101、S102を付与した各ステップの処理は図3と同じである。

【0110】

したがって、図7が図3と相違する点は、ステップS105だけである。

【0111】

ステップS105では、前記判定部108が、その時点で更新頻度記憶部107に記憶されている更新頻度Sに基づいてフレームページDP1上から主要領域を判定し、出力する。例えば、前記領域3に対応するHTMLファイルDP14の更新頻度Sが最も更新の頻度が高い値を有する場合には、当該HTMLファイルDP14が主要領域として選ばれ、出力されることになる。

【0112】

なお、必要ならば、更新の頻度が同じ値の領域が複数、存在する場合、前記有効期限情報などをもとに、1つの主要領域を特定するようにしてもよい。また、必要ならば、複数の領域を主要領域として出力したり、主要領域としては1つの領域を出力するが、更新の頻度が当該主要領域と同じ値を示す領域がほかに存在することをユーザU1に伝えるようにしてもよい。

【0113】

また、前記オートパイロットツールのようなソフトウェアに、前記日時や時間間隔のほか、前記URL11のようなフレーム規定ファイルを指定するためのURLを複数設定しておけば、同時進行で、多数のフレームページに対して同様な処理を行うことができる。これにより、多数のフレームページから主要領域だけを選択して出力させることが可能になり、ユーザU1は少ない労力と時間で、効

率的に、多くのフレームページの要点を認識することができる。

【0114】

ただしその場合には、分割結果記憶部104、更新頻度記憶部107はフレームページごとに整理して情報を記憶する必要がある。

【0115】

また、本実施形態では、特許文献1のように、ユーザU1が、データの開始箇所と終了箇所をあらかじめ手入力により指定する必要がないため、ユーザU1の操作負担が軽減できる。

【0116】

なお、本実施形態では、自動的に主要領域を判定できるので、(i)指定ウェブページの更新時の通知（主要領域以外の更新は通知しない、など）、(ii)検索（主要領域以外は検索対象としない、など）、(iii)要約（主要領域のみを要約対象とする、など）等のサービスやシステムを容易に構築することが可能となる。

【0117】

(A-3) 第1実施形態の効果

本実施形態によれば、フレームページから選択した主要領域を出力するためのユーザ(U1)の操作負担を軽減することが可能である。特に、多数のフレームページから主要領域だけを選択して出力させる場合に有効である。

【0118】

また、本実施形態は、基本的に自然言語処理を用いることなく実行可能であるため、記述言語に依存せずに主要領域を判定することができる。

【0119】

さらに、本実施形態では、フレームページの解析を実施するが、予め指定した文書構造だけを処理すればよいので、全ての解析をするよりも処理量が少ない。

【0120】

(B) 第2の実施形態

以下では、本実施形態が第1の実施形態と相違する点についてのみ説明する。

【0121】

第1の実施形態ではファイルの内容そのものを記憶し比較することで更新の有

無の検出などを行ったが、本実施形態では、ファイルの内容の替わりに、ファイルの内容のチェックサムを用いる。

【0122】

(B-1) 第2の実施形態の構成および動作

本実施形態と第1の実施形態は、前記領域処理部25の内部構成が相違するだけであるから、図15、図16、図17に示した構成は本実施形態でもそのまま用いることができる。本実施形態の領域処理部には符号35を付与して第1の実施形態の領域処理部25と区別する。

【0123】

領域処理部35の内部構成例は図12に示す通りである。

【0124】

図12において、当該領域処理部35は、読み込み部101と、バッファ部102と、分割部103と、更新頻度算出部106と、更新頻度記憶部107と、判定部108と、チェックサム算出部201と、チェックサム記憶部202と、チェックサム比較部203とを有する。

【0125】

このうち図2と同じ符号101～103、106～108、N10～N12、N17～N19を付与した各構成要素および各信号（各情報）の機能は第1の実施形態と同じであるので、その詳しい説明は省略する。

【0126】

また、前記チェックサム記憶部202は前記分割結果記憶部104に対応し、チェックサム比較部203は前記領域内容比較部105に対応する部分である。

【0127】

チェックサム記憶部202は、ファイルの内容（分割結果）の替わりに、チェックサム算出部201が当該ファイルの内容をもとに算出したチェックサムを示すチェックサム情報N30を記憶する部分である。ファイルの内容に比べると、通常、そのチェックサムははるかにサイズが小さいため、チェックサム記憶部202の記憶容量は、分割結果記憶部104の記憶容量よりも小さくてよい。

【0128】

したがって、チェックサム比較部203は、前回、読み込んだファイルのチェックサムと、今回、読み込んだファイルのチェックサムを比較することによって、各ファイルの更新の有無を検出し、検出結果に応じた更新有無情報N33を出力する。

【0129】

今回、読み込んだファイルのチェックサムを示すチェックサム情報N31も、今回、読み込まれバッファ部102内に格納されたファイルの内容をもとに、前記チェックサム算出部201が算出する。

【0130】

チェックサムをもとにファイルの更新の有無を検出すると、実際には更新されているのに更新されていないものとする誤検出が発生する可能性があるが、チェックサムのサイズ（ビット数）を長くすること等により、このような誤検出の発生確率を抑制することが可能である。

【0131】

本実施形態の動作を示すフローチャートは図7、図13、図14に示す通りである。このうち図7のフローチャートは、第1の実施形態と同じものを本実施形態でも用いる。

【0132】

また、図13のフローチャートは図3のフローチャートに対応し、同じ符号S101、S102を付与した各ステップの処理内容は第1の実施形態と同じであるので、その詳しい説明は省略する。

【0133】

図13において、ステップS201は図3のステップS103に対応し、ステップS202は図3のステップS104に対応する。

【0134】

ステップS201はS103と比べ、ファイルの内容ではなくチェックサムに基づいて更新の有無を検出する点が相違するだけである。

【0135】

また、ステップS202では、今回、読み込んだファイルのチェックサムを、

次回の新たなファイルの読み込みに備えて、前記チェックサム記憶部 2 0 2 に格納する。

【 0 1 3 6 】

一方、図 1 4 のフローチャートは図 8 のフローチャートに対応し、同じ符号 S 1 5 4 ~ S 1 6 0 を付与した各ステップの処理内容は第 1 の実施形態と同じであるので、その詳しい説明は省略する。

【 0 1 3 7 】

また、ステップ S 2 5 1 ~ S 2 5 3 でも、すでに述べたように、ファイルの内容の代わりにそのチェックサムを用いる点が第 1 の実施形態と相違するだけである。

【 0 1 3 8 】

(B - 3) 第 2 の実施形態の効果

本実施形態では第 1 の実施形態の効果と同等な効果を得ることができる。

【 0 1 3 9 】

加えて、本実施形態では、チェックサム記憶部 (2 0 2) に相当する部分の記憶容量が、第 1 の実施形態よりも少ないため、記憶資源を節約することができる。

【 0 1 4 0 】

また、チェックサムのサイズがファイルのサイズよりも小さければ、メモリなどの記憶資源に対する読み書きアクセスのための時間も短いため、処理時間を短縮できる可能性がある。

【 0 1 4 1 】

(C) 他の実施形態

上記第 1 の実施形態で、前回との比較を実施するために分割した結果を記憶しておくように構成したが、読み込んだ内容をそのまま記憶しておき、比較する前に改めて分割するようにしてもよい。

【 0 1 4 2 】

なお、分割した領域に対して合成ファイル S P 1 の記述の順番に従って対応付けを実施したが、フレームページの内部でそれぞれの領域を識別することができ

る情報（例えば、前記URL 1 2～URL 1 4）が付加されているので、その情報を使うように構成してもよいことはすでに述べた通りである。

【 0 1 4 3 】

また、上記第2の実施形態ではチェックサムを使うと説明したが、チェックサム以外の誤り検出方式に対応する符号を利用することもできる。また、ハッシュ関数を用いてファイルの内容を変換した値（ハッシュ値）を、前記チェックサムの代わりに利用するようにしてもよい。

【 0 1 4 4 】

なお、上記第1および第2の実施形態では、HTTPリクエストとしてGETリクエスト（GETメソッド）を用いる場合についてのみ説明したが、HEADリクエスト（HEADメソッド）を利用してもよい。

【 0 1 4 5 】

HEADリクエストに対するHTTPレスポンスは、エンティティボディ（ファイル）が含まれていないこと以外、GETリクエストに対するHTTPレスポンスと同じである。したがって通信端末1 2は、当該HEADリクエストを送信しても、前記WWWサーバ1 3から前記エンティティヘッダに含まれる有効期限情報や更新日情報などを取得することができる。

【 0 1 4 6 】

この場合、通信端末1 2では更新日情報などをもとに各ファイルの更新の有無を検出することが可能である。また、HEADリクエストによれば、サイズの大きなファイル本体を取り扱わなくて済むため、WWWサーバ1 3側でも通信端末1 2側でも負荷が小さく、通信トラフィックも少ない。また、レスポンスタイムも短いため、高速な処理が実行可能である。

【 0 1 4 7 】

さらに、更新日情報による更新の有無の検出と上記第1（または第2）の実施形態で述べたようなファイル本体（あるいは、そのチェックサム等）を比較することによる更新の有無の検出を混在させてもよい。例えば、ユーザU 1がフレームページの閲覧を希望してURL 1 1を入力した場合にはGETリクエストを用いて、ファイル本体を比較し、オートパイロットツールなどが自動的に更新の有

無を検出する場合などには、HEADリクエストを用いて、更新日情報による更新の有無の検出を行うようにしてもよい。

【0148】

なお、更新日情報などHTTPヘッダに含まれる各種情報は、WWWサーバ13側で収集された管理情報に基づいて記述されるため、WWWサーバ13側でファイルの管理が適正に行われていない場合などには、誤った情報となる可能性がある。例えば、実際にはまったく内容が変更されたいのに、ファイルの更新日が書き換えられ、更新が行われたかのような更新日情報が生成される可能性もある。

【0149】

ファイル本体の内容（やそのチェックサム等）を比較することによる更新の有無の検出などでは、このような誤りが混入しない点で、信頼性が高いといえる。

【0150】

なお、図4（A）～（D）の例では、トップページがフレームページになっているが、トップページ以外のページをフレームページとしてもよいことは当然である。

【0151】

また、フレームページの場合、図1に示したように、領域Bのような狭いフレームにメニューを置き、領域Cのような広いフレームにそのメニューの選択に応じて変わる内容を置く構成が典型例であるが、メニューのないフレームページにも、本発明が適用できることは当然である。

【0152】

さらに、フレームページ以外のWWWページに対しても本発明は適用可能であり、HTML以外の言語（XMLやSGMLなど）による記述にも適用可能である。何らかの意味で、論理的に識別可能な複数の領域が含まれていればよいからである。

【0153】

例えば、複数の画像（画像ファイル）を含むWWWページならば、各画像ファイルの更新頻度をもとにいずれかの画像ファイルを主要領域として判定してもよ

い。また、基本になるHTMLファイルとそのHTMLファイルに対応したWWWページ上に表示される画像（画像ファイル）のあいだで更新頻度をもとに主要領域を判定することも可能である。この場合、主要領域として画像（画像ファイル）が選択されると画像だけを表示し、HTMLファイルが選択されると画像を含まないHTMLファイルだけを表示してもよい。

【0154】

また、ファイル以外の単位をもとに領域を識別してもよいことは当然である。

【0155】

さらに、使用する通信プロトコルは必ずしもHTTPでなくともかまわない。

【0156】

なお、上記第1、第2の実施形態では通信端末（クライアント）12側に領域処理部25（または35）を配置したが、当該領域処理部の機能は、WWWサーバ13側に配置したり、WWWサーバ13と通信端末12の中間に介在し得る例えばプロキシサーバなどに配置することも可能である。

【0157】

WWWサーバ13側に配置した場合には、必ずしもHTTPによる通信を行わなくてもよい。また、WWWサーバ13のサーバOSが管理しているファイル管理の情報をそのまま活用し、更新の有無の検出などに利用することができる。

【0158】

また、第1および第2の実施形態では、フレームページがWWWサーバ13上に公開されたものであることを前提としているが、CD-ROMなどの記録媒体から得たフレームページ等にも本発明は適用できるので、対象とするフレームページ等は、必ずしもネットワーク経由で入手されるものでなくともかまわない。

【0159】

さらに、本発明で使用する式は前記式（1）に限定する必要はないことはすでに述べた通りである。例えば、更新されたことが検出されるたびに更新頻度の値がインクリメント（あるいは、デクリメント）されるような単純な式を利用することも可能である。

【0160】

以上の説明では主としてハードウェア的に本発明を実現したが、本発明はソフトウェア的に実現することも可能である。

【0161】

【発明の効果】

本発明によれば、構造化文書の中から主要な領域を判定するために必要なユーザの操作負担を軽減することができる。

【図面の簡単な説明】

【図1】

第1および第2の実施形態で使用する構造化文書の構成例である。

【図2】

第1の実施形態で使用する領域処理部の構成例を示す概略図である。

【図3】

第1の実施形態の「情報更新」時の動作を示すフローチャートである。

【図4】

第1の実施形態の動作説明に用いる構造化文書の例を示す説明図である。

【図5】

図4に示す構造化文書の各ファイルを読み込んだ結果の一例を示す説明図である。

【図6】

図5の構造化文書を分割した結果の一例を示す説明図である。

【図7】

第1の実施形態の「主要領域の内容出力」時の動作を示すフローチャートである。

【図8】

第1の実施形態の更新頻度Sの算出方法を示すフローチャートである。

【図9】

第1の実施形態の更新頻度Sの算出の具体例である。

【図10】

第1の実施形態の更新頻度Sの算出の具体例である。

【図 1 1】

第 1 の実施形態の「情報更新」時の動作と「主要領域の内容出力」時の動作とまとめた場合のフローチャートである。

【図 1 2】

第 2 の実施形態の領域処理部の構成例を示す概略図である。

【図 1 3】

第 2 の実施形態の「情報更新」時の動作を示すフローチャートである。

【図 1 4】

第 2 の実施形態の更新頻度 S の算出方法を示すフローチャートである。

【図 1 5】

第 1 および第 2 の実施形態の通信システムの全体構成例を示す概略図である。

【図 1 6】

第 1 および第 2 の実施形態で使用する通信端末の構成例を示す概略図である。

【図 1 7】

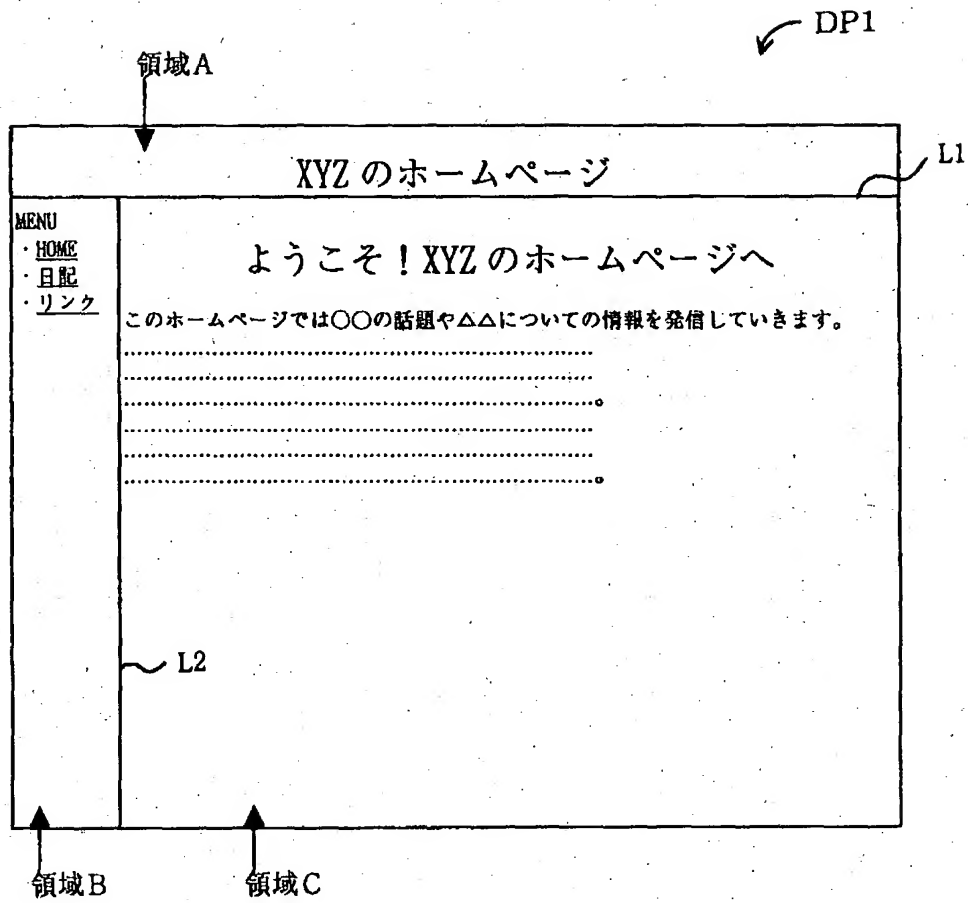
第 1 および第 2 の実施形態で使用する WWW サーバの構成例を示す概略図である。

【符号の説明】

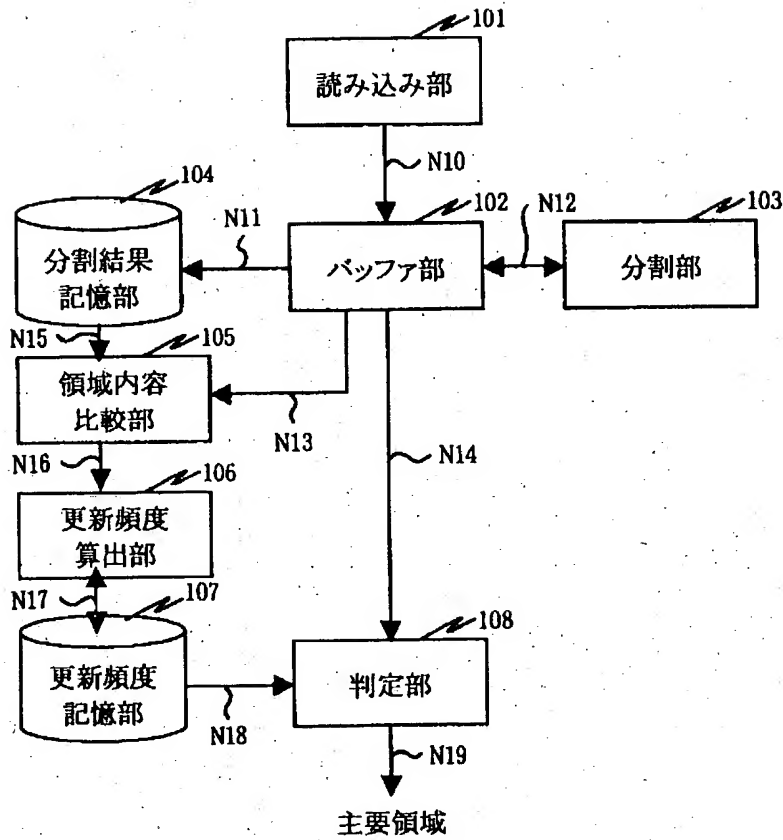
1 0 1 …読み込み部、1 0 2 …バッファ部、1 0 3 …分割部、1 0 4 …分割結果記憶部、1 0 5 …領域内容比較部、1 0 6 …更新頻度算出部、1 0 7 …更新頻度記憶部、1 0 8 …判定部、2 0 1 …チェックサム算出部、2 0 2 …チェックサム記憶部、2 0 3 …チェックサム比較部。

【書類名】 図面

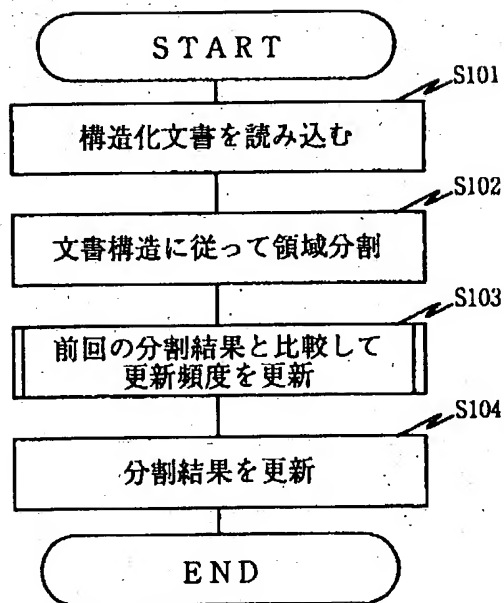
【図1】



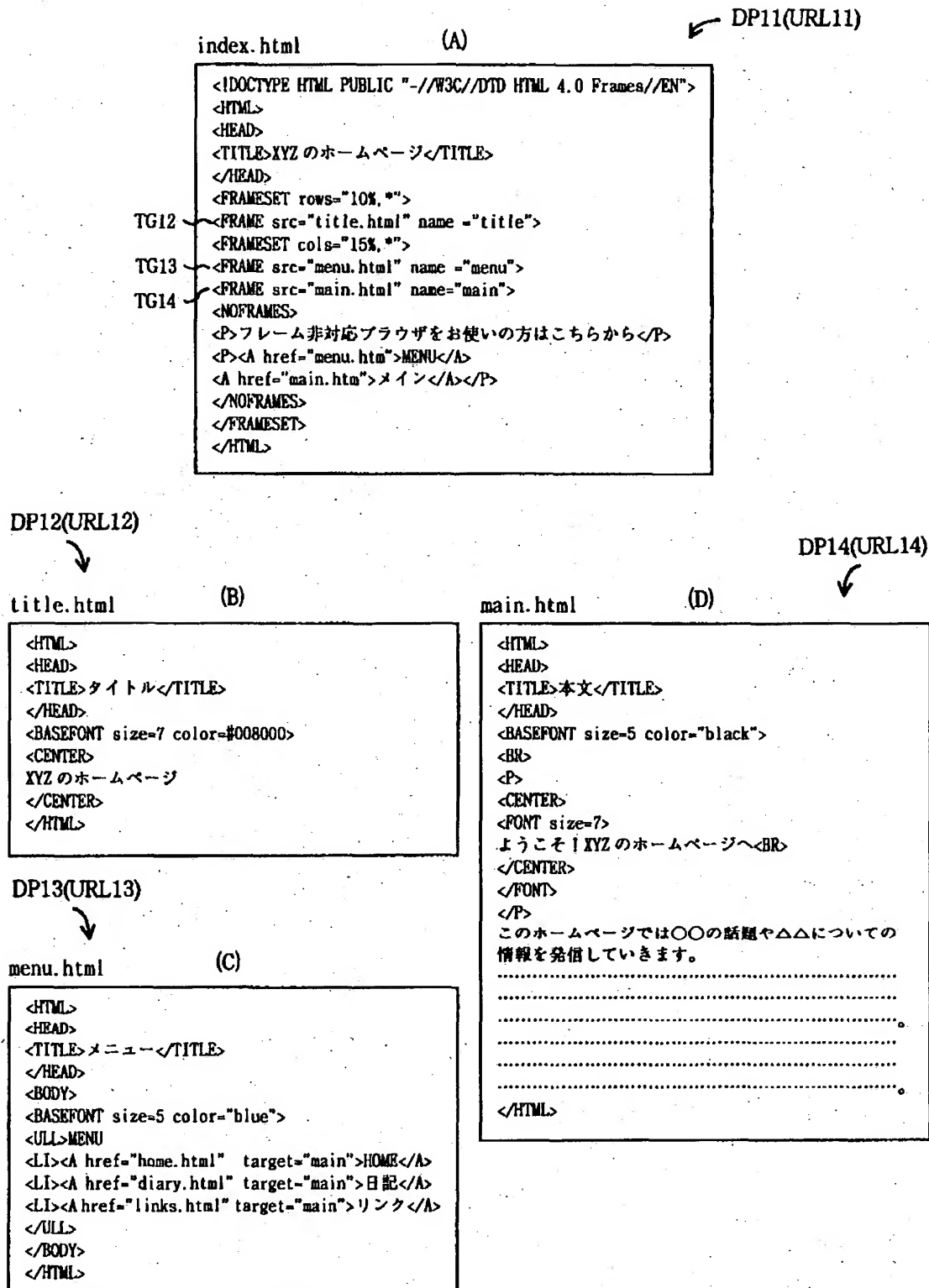
【図 2】



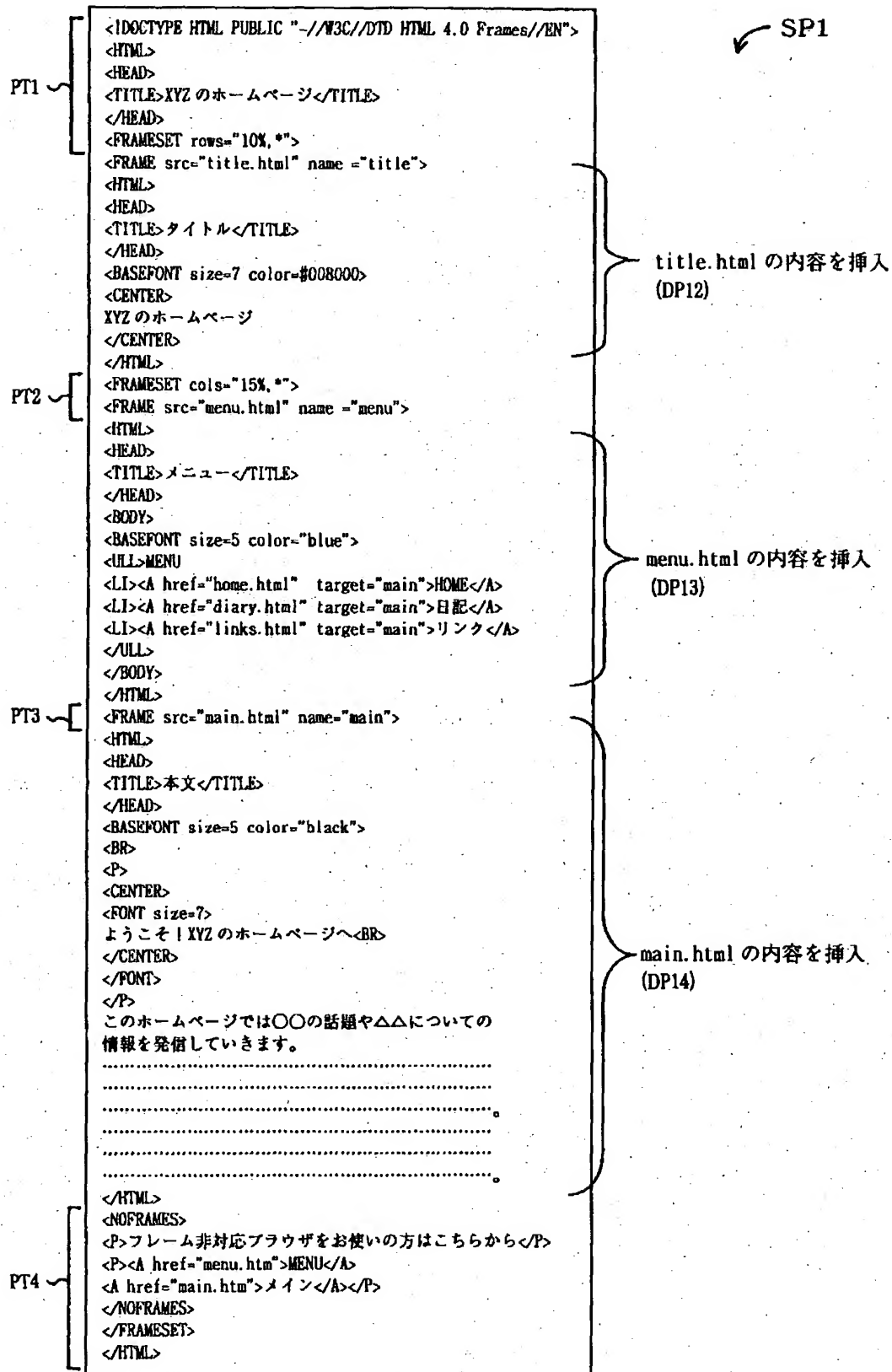
【図 3】



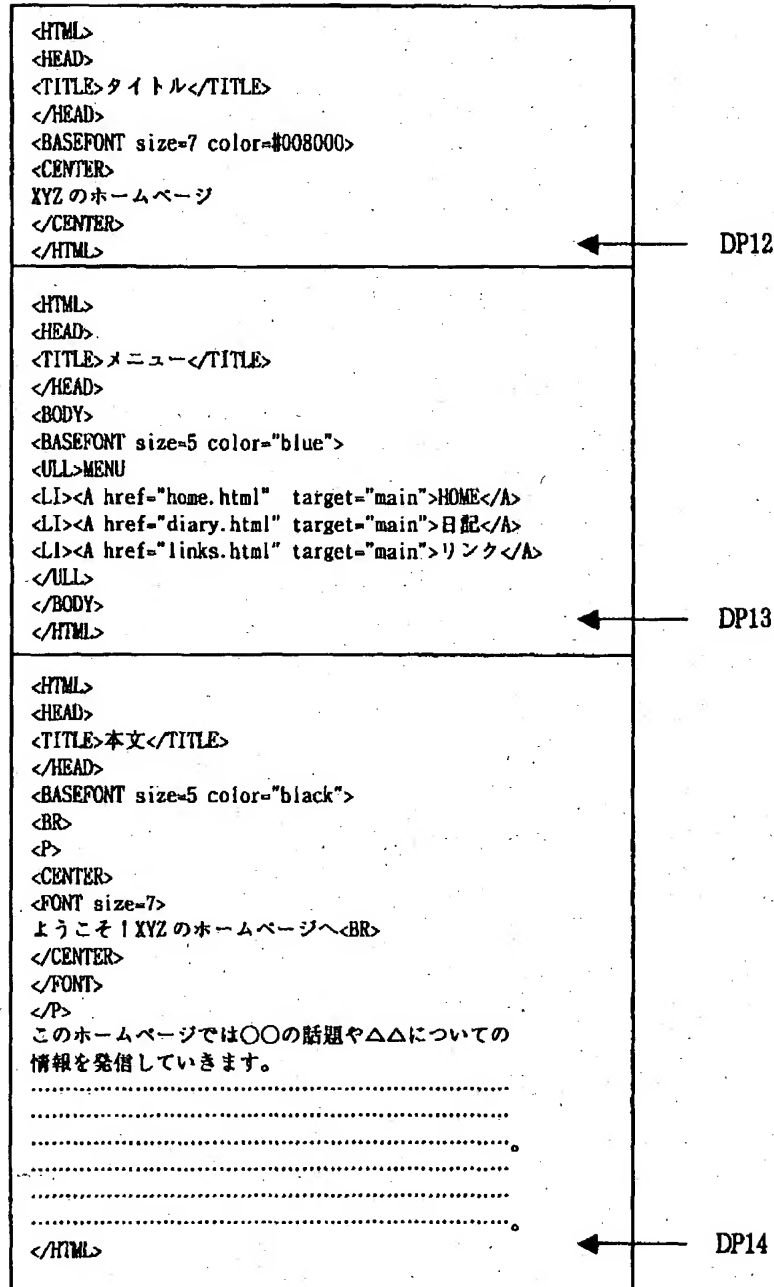
【図 4】



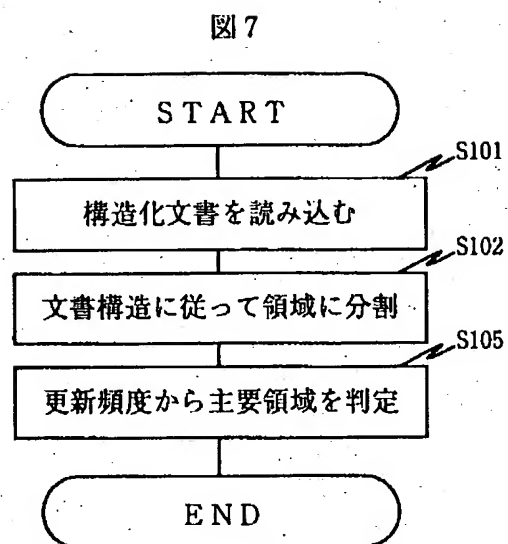
【図5】



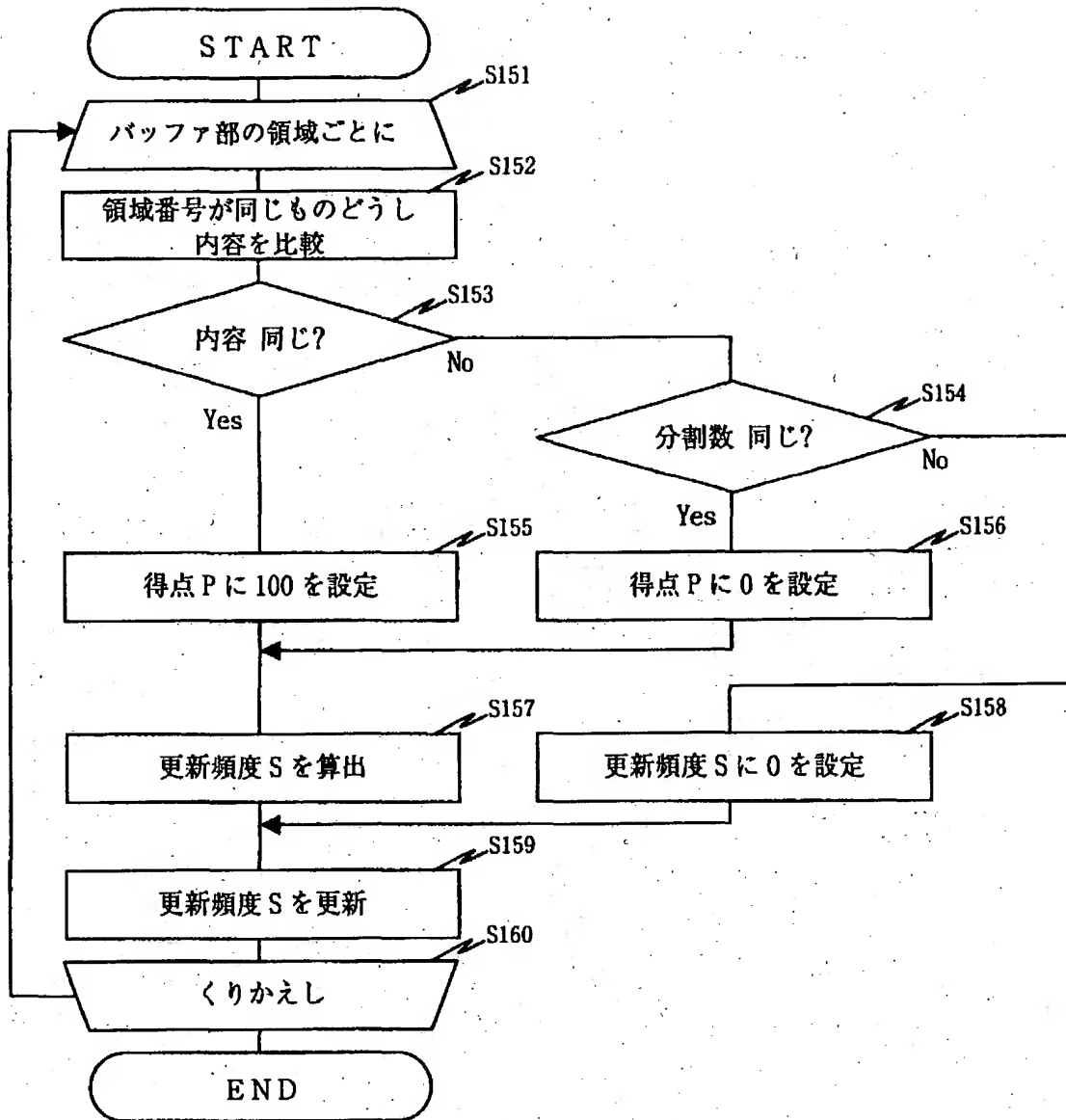
【図 6】



【図7】



【図 8】



【図 9】

	比較結果	得点 P	前回の 更新頻度 S_0	更新頻度 S
領域 1	同じ	100	73	78
領域 2	同じ	100	73	78
領域 3	違う	0	46	37

$\alpha = 0.8$ とすると

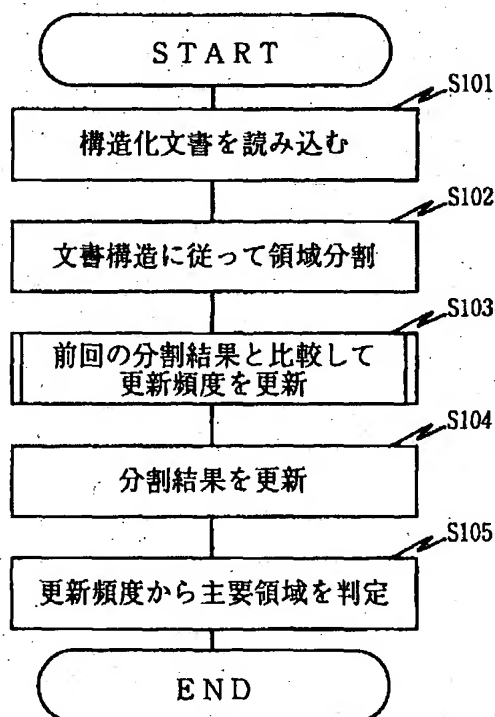
$$73 \times 0.8 + 100 \times (1 - 0.8) \doteq 78$$

$$46 \times 0.8 + 0 \times (1 - 0.8) \doteq 37$$

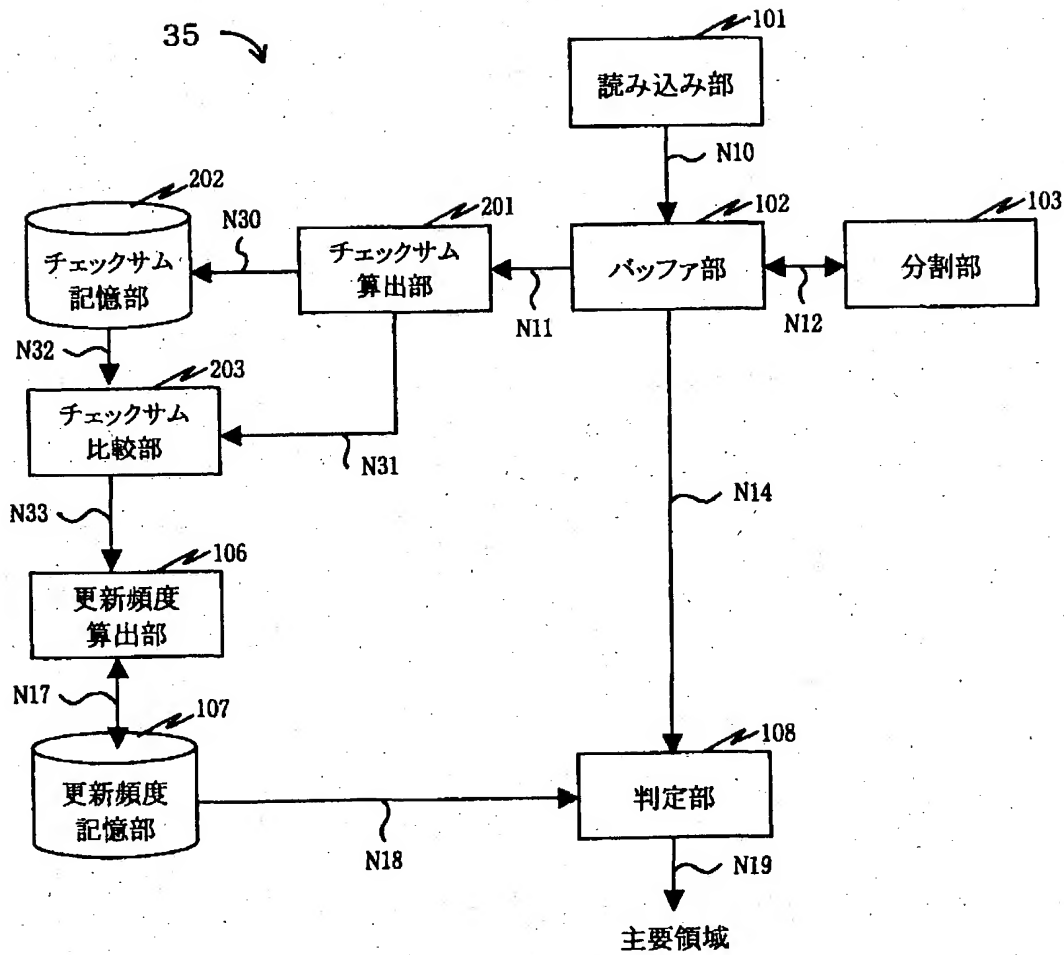
【図 10】

	比較結果	得点 P	前回の 更新頻度 S_0	更新頻度 S
領域 1	同じ	100	73	78
領域 2	違う	-	73	0
領域 3	違う	-	46	0
領域 4	比較対象なし	-	-	0

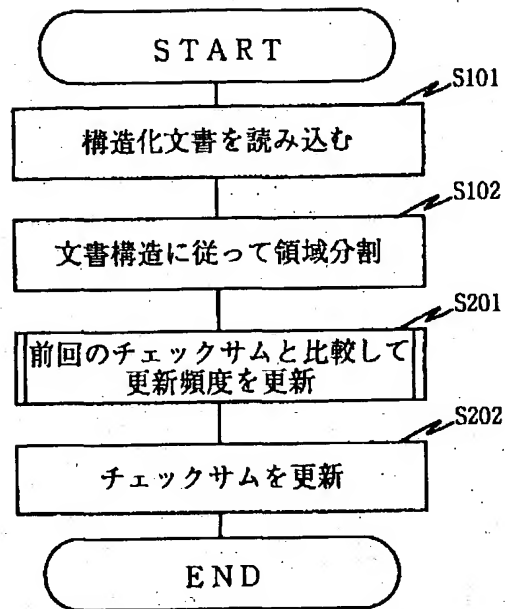
【図 11】



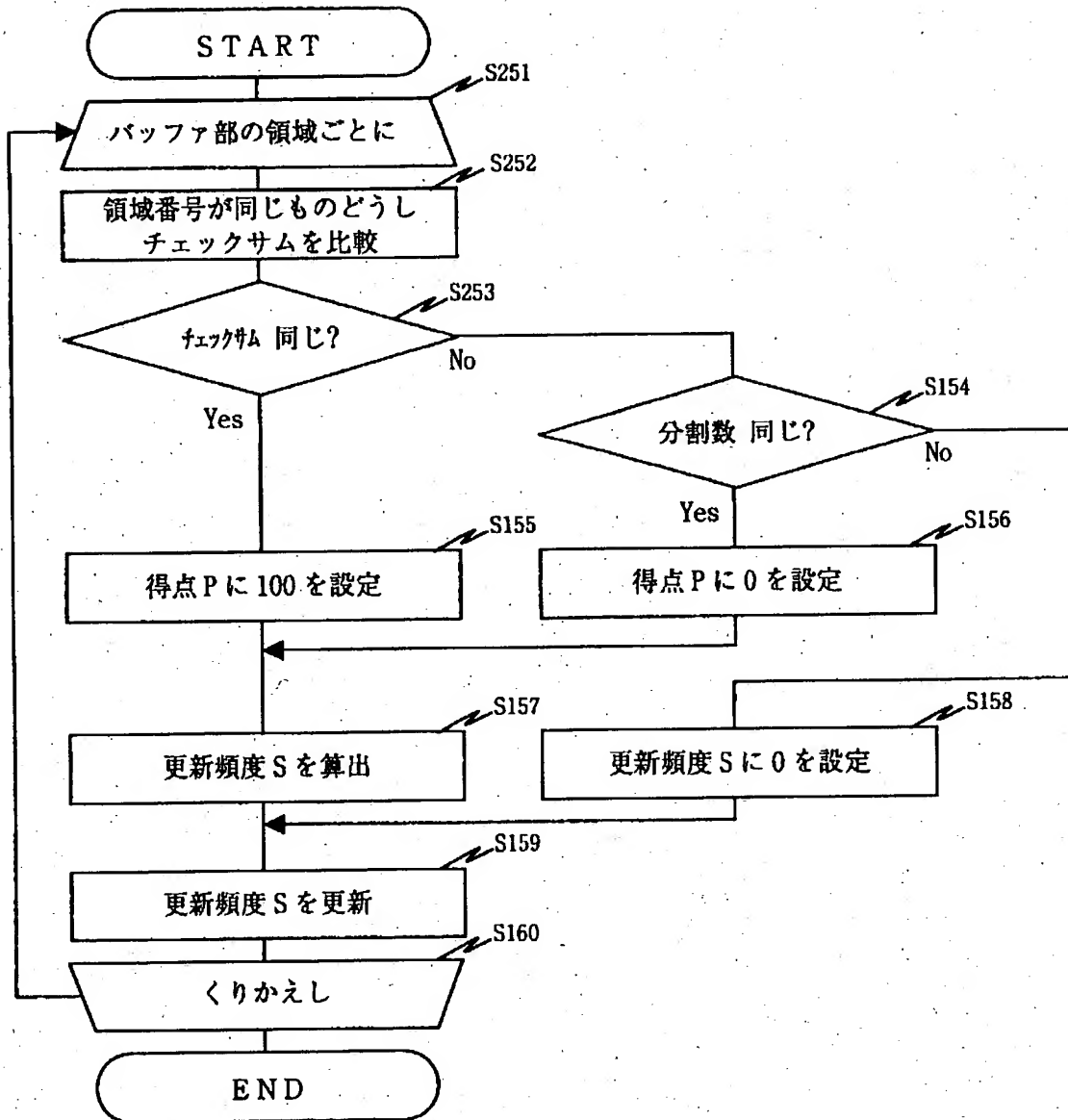
【図 12】



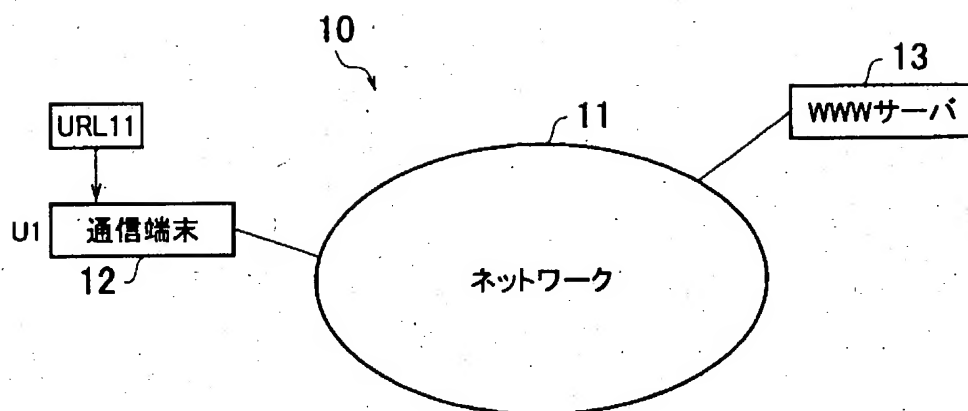
【図13】



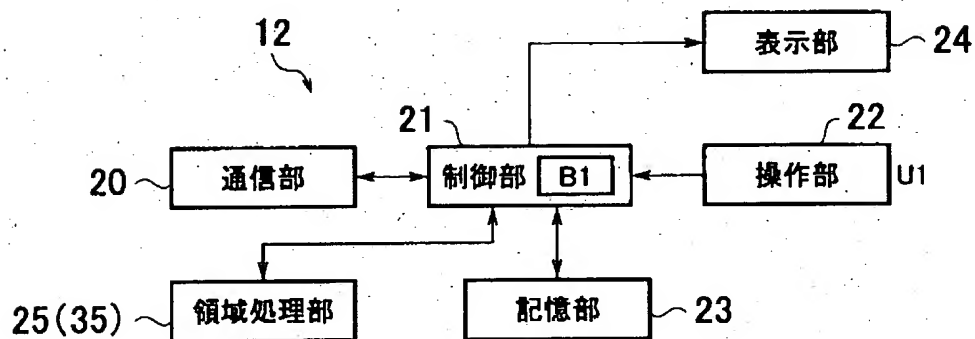
【図14】



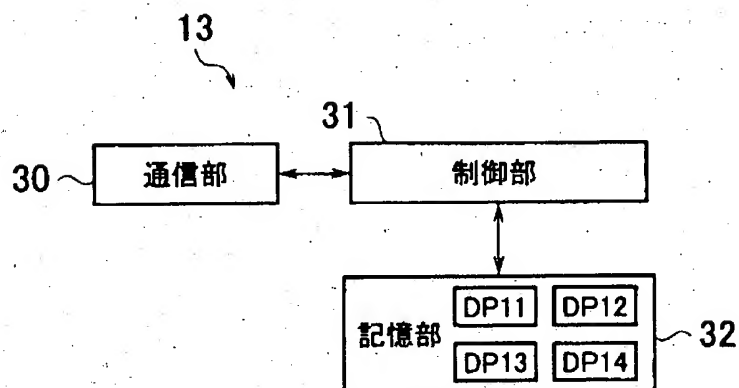
【図15】



【図16】



【図17】



【書類名】 要約書

【要約】

【課題】 操作負担を軽減する。

【解決手段】 複数の領域を含む所定の構造化文書の中から、主要な領域を判定する情報処理装置において、前記構造化文書中の各領域または、各領域の管理情報を時系列に複数回、取得する読み込み部と、当該読み込み部が取得した領域または管理情報を記憶する記憶部と、当該読み込み部が取得した対応する領域の間または管理情報の間で比較を行い、当該比較結果に基づいて各領域の更新の有無を検査する比較検査部と、当該比較検査部による検査結果の履歴をもとに、各領域ごとに、所定の更新頻度対応情報を算出する更新頻度算出部と、当該更新頻度対応情報に基づいて、前記構造化文書中の複数の領域の中から主要な領域を判定する主要領域判定部とを備える。

【選択図】 図 1

出 願 人 履 歴 情 報

識別番号 [000000295]

1. 変更年月日 1990年 8月22日

[変更理由] 新規登録

住 所 東京都港区虎ノ門1丁目7番12号

氏 名 沖電気工業株式会社